Beyond Next-Token Prediction: A Belief-State Architecture for AI

Nicholas Davidson

January 2026

Abstract

Abstract Modern artificial intelligence systems, particularly large language models, exhibit impressive surface-level competence across a wide range of tasks while remaining prone to inconsistency, hallucination, and brittle reasoning over time. These failures are often framed as issues of scale, data quality, or training methodology. In this paper, we argue that they arise from a more fundamental limitation: contemporary AI systems lack a persistent internal belief state that represents what the system takes to be true about the world across interactions. We propose that belief—defined as a structured, up-dateable, and enduring internal representation—constitutes a missing state variable in current AI architectures. We outline the properties such a belief state must possess, describe simple belief dynamics that govern how beliefs evolve in response to evidence, and situate this framework relative to Bayesian inference, Dempster–Shafer theory, and existing memory-augmented systems. Rather than proposing a new learning algorithm, this work offers a conceptual and architectural foundation for building AI systems that accumulate knowledge, maintain internal consistency, and reason more reliably over time.

1 Introduction

Recent advances in large language models have produced systems capable of generating fluent text, writing code, and performing complex multi-step reasoning. Despite these capabilities, persistent and well-documented failure modes remain. Models contradict themselves across sessions, fabricate unsupported claims, and struggle to maintain stable world models when reasoning over extended time horizons. These issues persist even as models grow larger and are trained on increasingly diverse datasets.

A common interpretation of these failures is that current systems are insufficiently scaled or imperfectly trained. From this perspective, hallucination and inconsistency are treated as engineering problems that will diminish as models become larger, datasets more comprehensive, and training procedures more refined. While continued progress along these axes has yielded incremental improvements, it has not eliminated the underlying issues.

This paper advances a different diagnosis. We argue that these failure modes are not incidental but structural. Modern AI systems are optimized to produce contextually plausible outputs, not to maintain a persistent internal representation of what

the system believes about the world. As a result, they can generate statements that resemble knowledge without possessing durable epistemic commitments. Each interaction is treated largely in isolation, and any apparent "memory" is confined to transient context windows or external retrieval mechanisms.

Human intelligence, by contrast, is characterized by the presence of beliefs that persist over time, are revised in response to evidence, and constrain future reasoning and action. These beliefs form an internal model of reality that evolves gradually rather than resetting with each new conversational turn. We contend that the absence of an analogous belief state in modern AI systems represents a fundamental architectural gap.

The contribution of this paper is conceptual rather than algorithmic. We introduce belief as a first-class computational state variable, specify the properties such a state must possess to support reliable reasoning, and describe simple dynamics by which beliefs can be updated, forgotten, and brought into consistency. We further argue that this framework is compatible with existing probabilistic inference methods while addressing limitations that arise when such methods are applied without a persistent cognitive substrate.

2 Belief as a Computational Primitive

In informal discussions of artificial intelligence, the term "belief" is often used loosely to refer to any information a system can reproduce. For the purposes of system design, such usage is insufficient. If belief is to play a functional role in artificial cognition, it must be defined as a concrete internal object with well-specified behavior.

We define a belief as a structured claim about the world, minimally characterized by a subject, a relation, and an object. For example, the statement "Paris is the capital of France" corresponds to a belief with subject Paris, relation $capital\_of$, and object $France$. Crucially, this claim is not represented merely as text, but as an internal record that carries additional state.

Each belief is associated with a confidence value indicating the system's degree of commitment, a timestamp reflecting when the belief was last updated, and provenance information describing the sources or evidence that support it. The collection of all such beliefs at a given time constitutes the system's belief state.

For belief to function as a cognitive primitive, it must satisfy several properties. Beliefs must persist across interactions rather than being recomputed from scratch. They must be up-dateable in light of new evidence, capable of decaying when no longer reinforced, and subject to consistency pressures when mutually incompatible claims arise. Finally, beliefs must constrain future reasoning, such that outputs that strongly contradict high-confidence beliefs are disfavored or explicitly flagged as uncertain.

In the absence of these properties, a system may generate statements that appear knowledgeable without maintaining any internal commitment to their truth. We argue that this distinction—between producing plausible text and maintaining belief—is central to understanding the limitations of current AI systems.

3 Why Next-Token Prediction Cannot Form Beliefs

Large language models are trained to estimate the conditional probability of the next token given a sequence of previous tokens. Formally, such models learn an approximation to

$$p(x_{t+1} \mid x_{\leq t})$$

where $x_{\leq t}$ denotes the token sequence observed so far. This objective has proven remarkably effective for learning linguistic structure and capturing broad statistical regularities in text. However, it places no direct pressure on a model to maintain persistent internal commitments regarding the truth of any particular proposition.

From the perspective of this paper, the central issue is not whether language models contain information about the world. Empirically, they clearly do. Rather, the issue is that this information is not organized as enduring belief state. The model's internal activations during inference are transient and are discarded once a response is produced. No explicit state representing what the system takes to be true is preserved across interactions.

This design has several consequences.

First, apparent knowledge is context-local. A model may generate a correct statement in one interaction and a contradictory statement in another without any internal mechanism registering the inconsistency. Because no belief persists beyond the immediate context window, there is no substrate upon which contradiction can be detected or resolved.

Second, the training objective does not reward epistemic consistency over time. Maximizing next-token likelihood encourages the production of text that is locally plausible, not text that remains globally consistent with previous outputs or with an internally maintained world model. A system can achieve high likelihood while producing mutually incompatible statements across different prompts, provided each statement is individually plausible in its local context.

Third, retrieval-augmented generation and long-context techniques do not address this limitation. External retrieval mechanisms supply additional information at inference time, but they do not create internal commitment. Retrieved content may influence a single response, yet it does not become part of a persistent belief state that subsequently constrains future reasoning. Similarly, increasing context length extends the horizon of local coherence but does not introduce durable memory or belief dynamics.

Fourth, fine-tuning and reinforcement learning from human feedback shape model behavior but still operate within the same stateless paradigm. These methods modify the parameters of the model so that certain outputs become more likely, but they do not endow the system with an explicit internal representation of beliefs that can be inspected, updated, or decayed over time.

Together, these observations imply that next-token prediction systems lack the representational machinery required for belief as defined in Section 2. They can approximate the surface form of knowledge without possessing knowledge in a functional sense. That is, they can generate statements that resemble beliefs without having any internal object corresponding to "this is something I believe."

This limitation is structural rather than incidental. Even a perfectly trained next-token predictor with infinite data and compute would remain a system whose internal state is reset at the start of each interaction. Without an additional persistent state variable representing belief, such a system cannot accumulate knowledge in the manner required for stable understanding or long-horizon reasoning.

We therefore conclude that the path to more reliable and coherent artificial intelligence does not lie solely in scaling next-token predictors. It requires the introduction of an explicit belief state and associated dynamics that operate alongside existing neural models. The next section describes a minimal framework for such belief dynamics.

4 Belief Dynamics

If belief is to function as a computational primitive, it must be possible to specify how beliefs evolve over time in response to evidence. In this section, we describe a minimal set of dynamics that endow beliefs with persistence, update-ability, and stability. The intent is not to propose a new probabilistic calculus, but to demonstrate that belief evolution can be expressed as a simple and well-behaved dynamical system.

4.1 Belief Representation

We assume that each belief b is associated with a scalar confidence value $w_t(b) \in [0, 1]$ at time t, representing the system's degree of commitment to that belief. A value of 0 corresponds to strong disbelief, while a value of 1 corresponds to strong belief. Intermediate values represent varying degrees of uncertainty.

The belief state at time t, denoted by B_t is the collection of all such belief-confidence pairs. This state persists across interactions and serves as the substrate upon which learning and reasoning operate.

4.2 Evidence as Support Signals

When the system encounters new information at time t, it is converted into evidence signals for relevant beliefs. For each belief b, the evidence produces a support value

$$s_t(b) \in [-1, 1]$$

where positive values indicate support, negative values indicate contradiction, and zero indicates irrelevance. Additionally, each piece of evidence is assigned a reliability value $\rho_t \in [0, 1]$ reflecting the trustworthiness of the source or observation.

These quantities need not be perfect; they may be produced by learning classifiers, heuristics, or other subsystems. The framework requires only that evidence be expressible as graded support.

4.3 Belief Update Rule

Belief updating is modeled as a gradual movement toward the confidence implied by new evidence. We first map support values to a target confidence:

$$\hat{w}_t(b) = \frac{1 + s_t(b)}{2}$$

Beliefs are then updated according to

$$w_{t+1}(b) = (1 - \eta_t) w_t(b) + \eta_t \hat{w}_t(b)$$

where $\eta_t = \alpha \rho_t$ and $\alpha \in (0, 1]$ is a learning-rate parameter.

This update rule has several desirable properties. Beliefs change smoothly rather than abruptly, preventing single observations from completely overturning high-confidence beliefs. Repeated consistent evidence drives convergence toward strong belief or disbelief. In the absence of new evidence, beliefs remain stable.

While this formulation resembles exponential smoothing, it may also be interpreted as a simplified form of probabilistic evidence accumulation. Alternative formulations, such as operating in log-odds space, are equally compatible with the framework.

4.4 Forgetting and Decay

Human beliefs weaken when they are not reinforced. To capture this phenomenon, we introduce a decay mechanism that gradually relaxes beliefs toward a neutral prior $\mu(b)$, often taken to be 0.5:

$$w_{t+1}(b) \leftarrow (1 - \lambda \Delta t) w_{t+1}(b) + (\lambda \Delta t) \mu(b)$$

where $\lambda \leq 0$ is a decay rate and Δt is the elapsed time since the belief was last updated.

Decay prevents the belief state from becoming clustered with obsolete or rarely used information and enables adaptation to changing environments.

4.5 Contradiction and Exclusivity

Certain sets of beliefs are mutually exclusive. For example, a given city can have only one capital-of relation. Let $\varepsilon(s, r)$ denote the set of beliefs sharing the same subject and relation but differing in object.

After individual belief updates, we enforce a normalization constraint:

$$w_{t+1}(b_i) \leftarrow \frac{w_{t+1}(b_i)}{\sum_{b_j \in \varepsilon(s,r)} w_{t+1}(b_j) + \epsilon}$$

This projection encourages competition among incompatible beliefs, ensuring that increasing confidence in one alternative decreases confidence in others. As a result, the system naturally drifts toward internally inconsistent world models.

4.6 Provenance

Each belief additionally stores provenance information describing the sources and evidence that contributed to its current confidence. Provenance is not required to participate directly in the update equations, but it enables downstream capabilities such as explanation, auditing, and selective trust in future evidence.

5 Architectural Implications

The introduction of a persistent belief state implies a corresponding shift in how artificial intelligence systems are organized. Rather than treating a single neural model as the locus of cognition, we propose a modular architecture in which neural networks serve primarily as perceptual and generative components, while belief state functions as a persistent internal substrate that mediates learning, reasoning, and action.

5.1 Separation of Perception and Belief

In the proposed architecture, neural models such as large language models, vision models, and speech models are responsible for transforming raw inputs into structured representations. These models excel at pattern recognition and linguistic fluency, but they do not themselves maintain long-term commitments about the world.

Belief state, by contrast, is a non-neural data structure that persists across interactions. It stores structured claims, associated confidences, and provenance, and it evolves according to the dynamics described in Section 4. This separation mirrors the distinction in biological systems between sensory processing and longer-term memory.

By decoupling perception from belief, the architecture allows neural models to be replaced, upgraded, or specialized without disrupting the agent's accumulated knowledge.

5.2 Evidence Extraction Layer

Between perception and belief state lies an evidence extraction layer. Its role is to convert model outputs or sensory observations into candidate beliefs and associated support signals. For example, given the sentence "Paris is the capital of France," the evidence extractor produces a belief candidate $(Paris, capital\_of, France)$ with positive support. Similarly, contradictory statements yield negative support. This layer may be implemented using learned information extraction models, entailment classifiers, or heuristic rules. Importantly, the framework does not require perfect extraction; belief dynamics are designed to tolerate noisy evidence and converge through repeated reinforcement.

5.3 Belief Store and Update Engine

The belief store maintains the persistent collection of beliefs. The update engine applies the dynamics described in Section 4 whenever new evidence arrives.

Together, these components form the system's internal world model. Unlike transient context buffers or retrieval systems, this world model represents the agent's own epistemic commitments and changes incrementally over time.

5.4 Belief-Constrained Reasoning and Generation

Belief state influences downstream behavior through constraint rather than replacement of neural generation.

When the system is asked to answer a question or perform a task, relevant beliefs are retrieved from the belief store and supplied to the reasoning or generation process.

Candidate outputs that strongly contradict high-confidence beliefs are penalized or flagged as uncertain.

This mechanism differs from simple retrieval augmentation. Retrieved beliefs are not treated as optional background context but as internal commitments that shape what the system considers acceptable. As a result, the system exhibits greater cross-session consistency and is less likely to produce unsupported claims.

5.5 Planning and Action

Although this paper focuses primarily on belief representation and dynamics, the same architecture naturally supports planning and action. A planner may treat belief state as a representation of the current world, propose actions, and predict their consequences. Observed outcomes then generate new evidence that updates beliefs.

This closed loop—belief informing action, action producing evidence, evidence updating belief—constitutes a minimal agentic cycle.

5.6 Summary Architecture

Conceptually, the system may be summarized as:

$$Perception \rightarrow EvidenceExtraction \rightarrow BeliefUpdate \rightarrow BeliefState$$

$$BeliefState \rightarrow Reasoning/Planning \rightarrow Generation/Action$$

Neural models operate at the periphery of this loop, while belief state occupies its center.

6 Relationship to Existing Frameworks

The framework proposed in this paper draws on ideas from probability theory, uncertainty reasoning, memory-augmented systems, and cognitive architectures. At the same time, it differs from each of these traditions in scope and intent. This section clarifies these relationships and situates belief-state architecture as a complementary and higher-level abstraction.

6.1 Bayesian Inference

Bayesian inference provides a principled method for updating the probability of a hypothesis given new evidence. It specifies how a prior belief should be combined with likelihood information to produce a posterior belief.

The present framework is Bayesian-compatible in the sense that belief update rules may be interpreted as approximate or simplified forms of Bayesian evidence accumulation. However, Bayesian inference alone does not define what constitutes a belief in a computational system, how beliefs persist over time, or how they interact with one another.

In particular, Bayesian inference assumes a predefined hypothesis space and operates as a local update rule. The framework proposed here instead defines a persistent belief state that evolves across interactions, supports creation and retirement of beliefs,

and enforces consistency constraints. Bayesian inference may be employed within this state, but it does not substitute for the architecture itself.

6.2 Dempster-Shafer Theory

Dempster-Shafer theory generalized probabilistic reasoning by allowing belief mass to be assigned to sets of hypotheses and by distinguishing between belief and plausibility. It is primarily concerned with combining evidence from multiple uncertain sources.

As with Bayesian inference, Dempster-Shafer theory addresses the problem of evidence combination rather than the problem of cognitive organization. It does not specify how beliefs persist, how they constrain reasoning, or how they guide action over time. Within the proposed framework, Dempster–Shafer mechanisms may be used as an implementation choice for belief updating, but the contribution of this paper lies in defining the surrounding belief-state architecture.

6.3 Retrieval-Augmented Generation

Retrieval-augmented generation (RAG) systems incorporate external documents or memories into the generation process by retrieving relevant information at inference time. While such systems improve factual accuracy in many cases, they do not establish internal commitment to retrieved information.

Retrieved content influences only the current response and does not become part of a persistent internal belief state that subsequently constrains behavior. In contrast, the present framework treats retrieved or observed information as evidence that updates internal beliefs, thereby affecting future interactions even in the absence of repeated retrieval.

6.4 Knowledge Graphs

Knowledge graphs represent facts as structured triples and are widely used for storing and querying information. However, traditional knowledge graphs are largely static and lack intrinsic mechanisms for uncertainty, evidence-based updating, or temporal evolution.

The belief-state framework may be viewed as extending the idea of a knowledge graph with confidence values, provenance, and explicit dynamics. Beliefs are not merely stored but continuously revised in response to evidence.

6.5 Memory-Augmented Neural Networks

A variety of architectures augment neural networks with external memory modules, enabling storage and retrieval of information across time. These systems demonstrate that memory can improve performance on certain tasks, but memory entries are typically unstructured or treated as latent vectors.

The present framework differs in that memory is structured as explicit beliefs with defined semantics and update rules. Rather than learning memory access patterns end-to-end, the system maintains an interpretable belief store governed by explicit dynamics.

6.6 Cognitive Architectures

Classical cognitive architectures such as SOAR and ACT-R attempt to model aspects of human cognition using symbolic representations and rule-based systems. These architectures emphasize structured knowledge and reasoning but are largely disconnected from modern large-scale neural perception models.

The proposed framework can be seen as a synthesis: it combines the representational clarity of symbolic belief systems with the perceptual capabilities of contemporary neural networks. Unlike prior cognitive architectures, it is designed specifically to operate alongside large language models and other deep learning systems.

7 Predictions and Evaluation

Although this paper is primarily conceptual, it makes concrete predictions about the behavior of systems that incorporate a persistent belief state. These predictions distinguish belief-state architectures from purely stateless next-token prediction systems and suggest empirical tests that can be conducted in controlled environments.

7.1 Behavioral Predictions

A system equipped with a persistent belief state and belief dynamics as described in Sections 2-5 should exhibit the following properties:

1. **Cross-Session Consistency**
   The system should produce consistent answers to factual queries across separate interactions, even when not explicitly reminded of prior statements.
2. **Belief Revision**
   When presented with credible contradictory evidence, the system should gradually revise its beliefs rather than oscillating or rationalizing inconsistencies.
3. **Contradiction Awareness**
   The system should be able to identify when new information conflicts with high-confidence beliefs and either lower confidence or express uncertainty.
4. **Reduced Confabulation**
   The system should be less likely to assert unsupported claims, preferring to express uncertainty when relevant beliefs are absent of low-confidence.
5. **Accumulation of Experience**
   Performance should improve over time within a fixed environment as beliefs accumulate and stabilize.

These properties are not guaranteed by scaling stateless language models alone.

7.2 Toy Environments

Evaluation may be conducted in small, controlled environments designed to require persistent knowledge. Examples include:

- A simulated world containing a limited number of entities and relations that change over time.

- A text-based detective environment in which clues accumulate gradually.
- A simple object-manipulation environment with hidden state.

Such environments allow ground-truth tracking of world state and objective measurements of belief accuracy.

7.3 Baselines

Belief-state agents should be compared against:

- A stateless language model
- A language model with retrieval-augmented generation
- A language model with long context but no persistent belief store

All systems should have access to the same perceptual inputs.

7.4 Metrics

Potential evaluation metrics include:

- **Belief Accuracy:** agreement between internal beliefs and ground truth
- **Consistency Rate:** probability of giving the same answer across sections
- **Belief Update Accuracy:** correctness of revisions following new evidence
- **Contradiction Rate:** frequency of mutually incompatible assertions
- **Task Success Rate:** completion of environment-specific objectives

These metrics directly target the claims of the framework.

7.5 Ablation Studies

Ablations may include:

- Removing decay
- Removing contradiction constraints
- Replacing belief update rules with naive overwriting

Such studies help isolate which components contribute most to observed improvements.

8 Limitations and Future Work

The framework proposed in this paper is intentionally minimal. Its purpose is to introduce belief as a computational primitive and to outline a class of architectures in which belief state plays a central role. Several important limitations remain.

First, evidence extraction and scoring are nontrivial. The framework assumes that incoming observations can be converted into graded support signals for candidate beliefs. In practice, this process will be noisy and imperfect, particularly when operating on unstructured text. Improving the reliability of evidence extraction is a significant open problem.

Second, the belief schema adopted here is deliberately simple. Representing beliefs as subject–relation–object triples captures many factual claims but may be insufficient for complex relational, temporal, or procedural knowledge. Extending the schema to accommodate richer forms of belief while preserving tractable dynamics is an important direction for future work.

Third, the update rules described in Section 4 are hand-specified. While they are simple and well-behaved, it may be beneficial to learn portions of the update dynamics from data or interaction. Hybrid approaches that combine fixed structure with learned parameters represent a promising avenue.

Fourth, this paper does not address large-scale deployment, efficiency, or engineering constraints. The intent is to establish conceptual foundations rather than to propose a production-ready system. Scaling considerations, including memory management, distributed belief stores, and real-time performance, are left for future investigation.

Finally, persistent belief state raises important safety and alignment questions. Systems that accumulate long-term internal models may develop undesirable or biased beliefs if exposed to skewed data. Mechanisms for auditing, correcting, and constraining belief formation will be essential.

Despite these limitations, the framework opens multiple research directions, including belief-driven planning, causal belief formation, hierarchical belief representations, and integration with reinforcement learning.

9 Conclusion

Modern artificial intelligence systems exhibit impressive surface competence while lacking stable internal representations of what they take to be true. This paper argues that the absence of a persistent belief state constitutes a fundamental architectural gap.

We introduced belief as a first-class computational primitive, described minimal dynamics governing belief evolution, and outlined an architecture in which belief state mediates learning, reasoning, and action. Rather than proposing a new model family or training algorithm, this work reframes the problem of intelligence at the level of system structure.

We contend that progress toward more reliable and coherent artificial intelligence will require moving beyond purely stateless next-token prediction toward architectures that maintain and revise internal models of the world. Belief state provides one such foundation.